Subject: Object-Oriented Programming
Code: 18506
Institution: Escuela Politécnica Superior
Degree: Telecommunication Technologies and Services Engineering Degree
Level: Graduate
Type: Elective
ECTS: 6

COURSE TITLE

OBJECT-ORIENTED PROGRAMMING (OOP)

1.1.   Course number

18506

1.2.   Course area

Computer Systems

1.3.   Course type

Elective

1.4.   Course level

Graduate

1.5.   Year

3º

1.6.   Semester

2º

1.7.   Credit allotment

6

1.8.   Prerequisites

OOP belongs to Subject "Computer Systems", 18 ECTS, from the elective subjects of the Degree of Telecommunication Technologies and Services Engineering Degree. No prerequisites are needed to attend this course, although it is highly recommended that the student has intermediate knowledge about the C language.

Subject: Object-Oriented Programming
Code: 18506
Institution: Escuela Politécnica Superior
Degree: Telecommunication Technologies and Services Engineering Degree
Level: Graduate
Type: Elective
ECTS: 6

1.9.   Minimum attendance requirement

1.10.   Faculty data

Add @uam.es to all email addresses below.

Theory:

Dr. Francisco Saiz López (Coordinator)
Departamento de Ingeniería Informática
Escuela Politécnica Superior
Office: B-323 Edificio B – 3ª Planta
Tel.: +34 91 497 2281
e-mail: francisco.saiz
Web: http://www.ii.uam.es/~saiz

Practice:

To be defined (Coordinator)
Departamento de Ingeniería Informática
Escuela Politécnica Superior
Office: B-323 Edificio B – 3ª Planta
Tel.: +34 91 497 2281
e-mail: francisco.saiz
Web: http://www.ii.uam.es/~saiz

1.11.   Course objectives

Students will learn to analyze, design, implement and test software using the paradigm of object-oriented programming. The student will be able to use methods, engineering tools and techniques for building robust applications and maintainable by using this paradigm. Special emphasis will be on the implementation phase, using the Java programming language.

The specific ability to Computer Science that the student should acquire with this course is:

- FB2. Basic knowledge on using and programming computers, operating systems, databases and software with applications in engineering.

Subject: Object-Oriented Programming
Code: 18506
Institution: Escuela Politécnica Superior
Degree: Telecommunication Technologies and Services Engineering Degree
Level: Graduate
Type: Elective
ECTS: 6

3 de 4

The results of the learning the student acquires are:

- Understanding the fundamentals of object-oriented technology. Ability to design and implement solutions by using that technology and their application in specific areas such as user interfaces.
- Ability to implement solutions using the object-oriented paradigm: programming languages and methods of analysis, design, integration, and testing of object-oriented software.

At the end of each unit, the student should be capable of:

UNIT BY UNIT SPECIFIC OBJECTIVES	
TEMA 1.- Introduction to OOP	
1.1.	Revise methodologies of development
1.2.	Show UML diagrams
TEMA 2.- Classes in Java	
2.1.	Introduce imperative sentences in Java
2.2.	Include objects and classes as elementary building blocks
2.3.	Study methods of encapsulating data
2.4.	Present the mechanisms of modularity
TEMA 3.- Class inheritance	
3.1.	Present class hierarchy
3.2.	Study abstract classes and interfaces
3.3.	Introduce error handling by means of exceptions
3.4.	Present generics and inner classes
3.5.	Study polymorphism and concurrency
TEMA 4.- Design of the user interface	
4.1.	Present principal GUI elements
4.2.	Incorporate layout strategies
4.3.	Study event management and listeners
4.4.	Study the Swing framework
4.5.	Present graphics elements
TEMA 5.- Design patterns	
5.1.	Study general concepts on patterns
5.2.	Present some representative examples in Java

1.12. Course contents

1. **Introduction to OOP**
   1.1. Software development phases. Methodologies.
   1.2. Design diagrams. UML.
2. **Classes in Java**
   2.1. Imperative programming
   2.2. Objects and classes
   2.3. Encapsulation
   2.4. Modularity
3. **Class inheritance**
   3.1. Class hierarchy. Encapsulation.
   3.2. Abstract classes and interfaces
   3.3. Error handling. Exceptions.
   3.4. Collections and generics. Inner classes.
   3.5. Polymorphism. Concurrency.
4. **Design of the user interface**
   4.1. GUI elements
   4.2. Geometric distribution of components. Layouts.
   4.3. Event Management. Listeners.
   4.4. The Swing framework.
   4.5. Using graphics
5. **Design patterns**
   5.1. General concepts of patterns
   5.2. Representative examples in Java

1.13. Course bibliography

1. http://docs.oracle.com/javase/tutorial (Java tutorial)
2. "Core Java 2 Vol. 1 Fundamentos", Horstmann, C. S. Ed. Prentice Hall, 2006.
3. INF/681.3.062-J/HOR Vol. 1
4. "Core Java 2 Vol. 2 Características avanzadas", Horstmann, C. S. Ed. Prentice
5. Hall, 2006. INF/681.3.062-J/HOR Vol. 2
6. "Java programming", Farrell, J. Ed. Course Technology, 2012. INF/681.3.062-J/FAR
7. http://docs.oracle.com/javase/tutorial/uiswing (Java Swing tutorial)
8. "Swing a beginner's guide", Schildt, H. Ed. McGraw Hill, 2007. INF/681.3.06-S/SCH
9. "Patrones de diseño elementos de software orientado a objetos reutilizable", Gamma, E., Helm, R., Johnson, R., Vlissides, J. Ed. Wesley, 2003. INF/681.3.06/PAT